US006279047B1

(12) **United States Patent**
Bublitz et al.

(10) **Patent No.: US 6,279,047 B1**
(45) **Date of Patent: Aug. 21, 2001**

(54) **METHOD FOR SIMPLIFYING COMMUNICATION WITH CHIP CARDS**

(75) Inventors: **Hermann Bublitz**, Böblingen; **Klaus Rindtorff**, Weil i. Schönbuch, both of (DE)

(73) Assignee: **International Business Machines Corporation**, Armonk, NY (US)

(*) Notice: This patent issued on a continued prosecution application filed under 37 CFR 1.53(d), and is subject to the twenty year patent term provisions of 35 U.S.C. 154(a)(2).

Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **08/793,911**

(22) PCT Filed: **Jun. 7, 1996**

(86) PCT No.: **PCT/EP96/02483**

§ 371 Date: **Jun. 13, 1997**

§ 102(e) Date: **Jun. 13, 1997**

(87) PCT Pub. No.: **WO97/01147**

PCT Pub. Date: **Jan. 9, 1997**

(30) **Foreign Application Priority Data**

Jun. 23, 1995 (DE) 195 22 527

(51) **Int. Cl.**[7] **G06F 13/10**

(52) **U.S. Cl.** **710/11**; 710/8; 710/10; 710/36; 710/102; 710/104; 710/105

(58) **Field of Search** 463/45; 340/825.3; 327/415; 235/492; 711/1, 2, 5, 115, 170, 171, 172, 173, 201; 710/8, 10, 11, 13, 17, 101, 102, 103, 104, 105; 709/328, 329

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,870,704	*	9/1989	Matelan et al.	710/120
4,874,935	*	10/1989	Younger	235/492
5,327,018	*	7/1994	Karlish et al.	327/415
5,517,646	*	5/1996	Piccirillo et al.	713/1
5,534,855	*	7/1996	Shockley et al.	340/825.3
5,590,193	*	12/1996	Le Roux	380/4
5,624,316	*	4/1997	Roskowski et al.	463/45
5,983,288	*	11/1999	Visee	710/16

* cited by examiner

Primary Examiner—B. James Peikari
(74) *Attorney, Agent, or Firm*—Karl O. Hesse

(57) **ABSTRACT**

An apparatus according to the invention for communication of an application (**10**) with a chip card (**20**) in which there is at least one application data storing means (**30**), also known as the dictionary, for recording of information concerning application specific data of the application (**10**), and at least one chip card dialog module (**40**), also known as the agent, for the generation of commands using the application data storing means (**30**) for an interface to the chip card (**20**), whereby the chip card dialog module (**40**) contains card specific data concerning the chip card (**20**). A request (**100**) for communication to the chip card (**20**) is made by the application (**10**) to the chip card dialog module (**40**) which is specified for the chip card (**20**). Upon the request (**100**), the chip card dialog module (**40**) generates at least one command (**110**), which is required for communication with the chip card (**20**). To achieve this, the chip card dialog module (**40**) accesses application specific information for itself which is stored in the application data storing means (**30**).

26 Claims, 2 Drawing Sheets

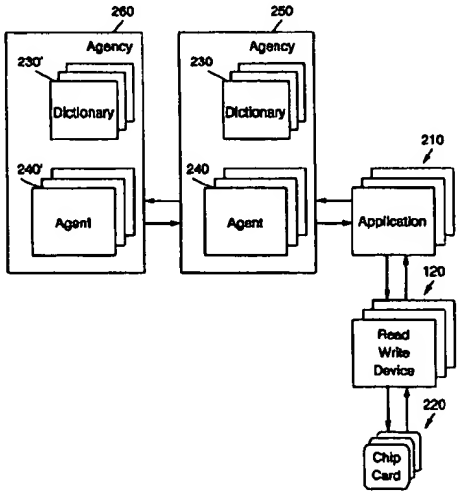
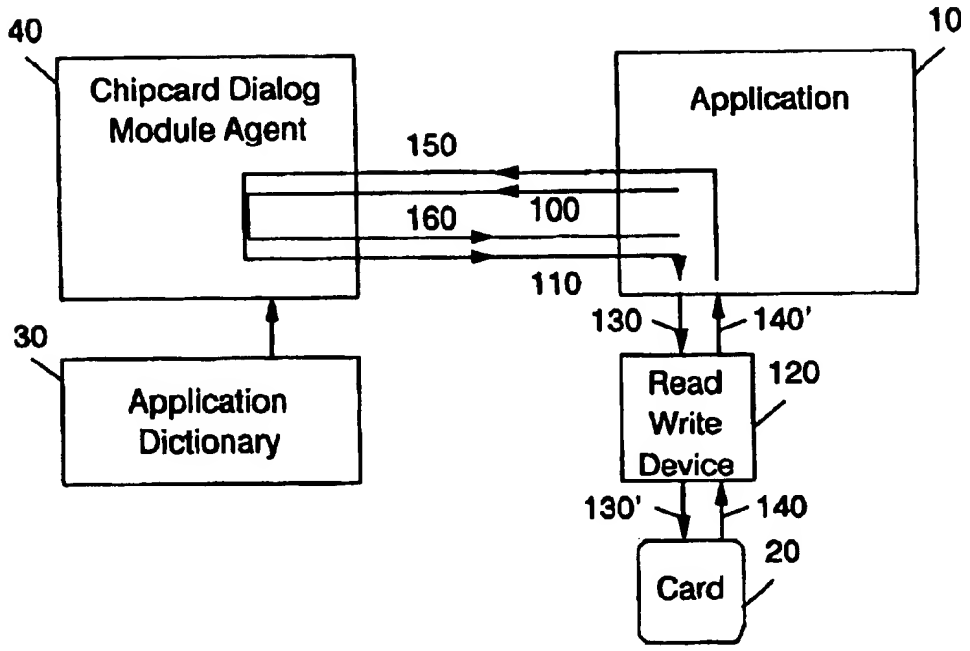


FIG. 1


40
Chipcard Dialog Module Agent
10
Application
150
160
100
110
130
140'
30
Application Dictionary
Read Write Device
120
130'
140
20
Card

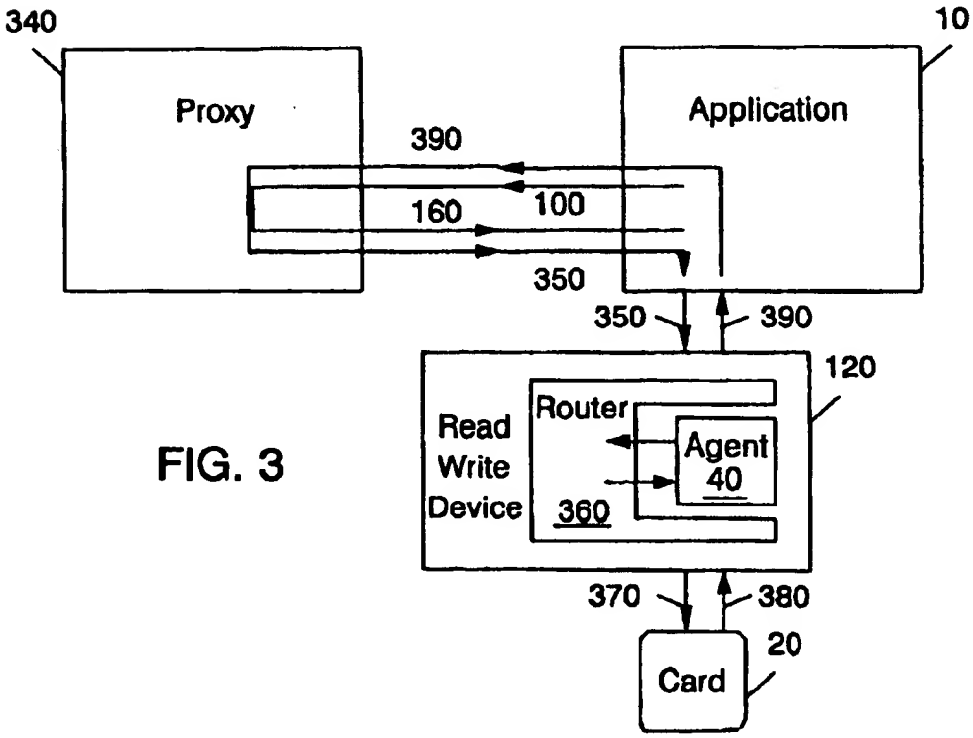

340
Proxy
10
Application
390
160
100
350
350
390
120
Read Write Device
Router
Agent 40
360
370
380
20
Card

FIG. 3

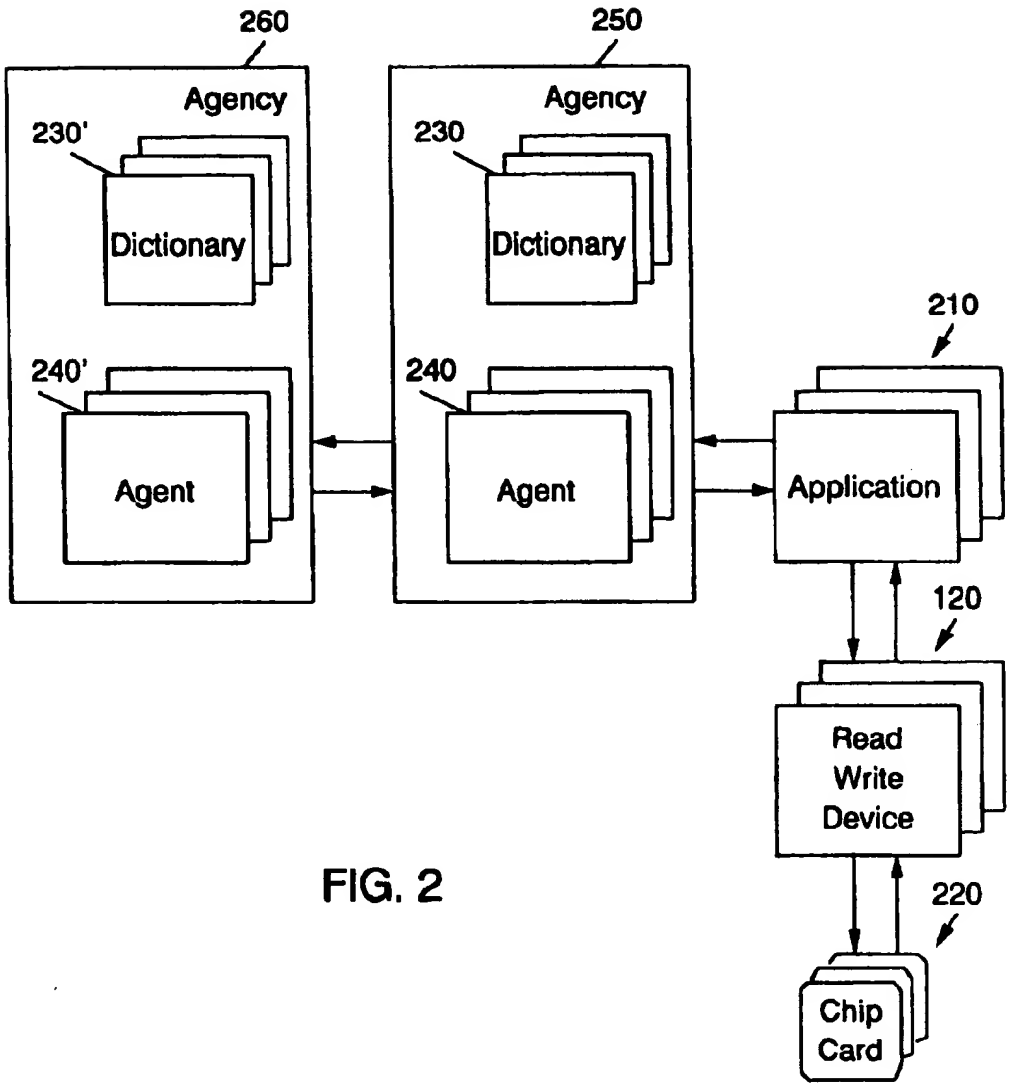

260
Agency
230'
Dictionary
240'
Agent
250
Agency
230
Dictionary
240
Agent
210
Application
120
Read
Write
Device
220
Chip
Card

FIG. 2

METHOD FOR SIMPLIFYING COMMUNICATION WITH CHIP CARDS

TECHNICAL FIELD

The invention relates to an apparatus and a method for the communication of an application with a chip card, as well as a use in reading/writing devices for chip cards.

DESCRIPTION OF RELATED ART

Cards with an integrated electronic chip which are efficient, generally small, roughly in the size and shape of a credit card, and made preferably of plastic or metal, are known today as data carrier cards or chip cards. In contrast to simple storage cards (which are known as memory chip cards or memory cards), intelligent data carrier cards (also known as multifunctional chip cards or smart cards) possess, in addition to their memory function, their own processor for control of the data stored on the chip of the data carrier card. This makes better protection of the data possible, and results in improved functioning of the chip cards. In general, simple memory cards allow only writing and reading of data. Smart cards, in addition to these functions, also possess functions for structuring of the data, for isolating the data, for administration of the data and for protection of the data. The interface of a smart card, and thus the programming required in order to read data from the chip card (of command sequences, for example), is therefore significantly more complex than that of memory cards.

The installation of chip cards, which is growing primarily due to their enhanced security against tampering, extends to a wide range of application areas. Applications with chip cards, that is, applications in which the communication of any random device with a chip card is required, may be for making payments without cash, identification of the chip card holder, the storage of data, or other similar uses. An application of this type consists of internal application segments on the chip card and external application segments in corresponding devices such as automatic cash machines, PC's or special terminals. In general, internal application segments represent all of the data and programs which are in memory on the chip card itself, while the external application segments represent all of the data and programs outside of the chip card. Programming of these external application segments is typically the responsibility of those who program the devices with which a chip card is to be installed. Frequently, these devices have a complex software platform of their own. In order to integrate applications with chip cards in these instances, extensive knowledge of the structure of the stored data and the interface of the chip card is required.

Most of the chip cards used differ widely among themselves from manufacturer to manufacturer. Many of the chip cards implement a partial set of ISO Standard 7816, and also make use of several special functions. In addition, there is in actual practice the required knowledge of the details of the internal and external application segments to be implemented.

In order to enable access to the data stored on chip cards, a fixed programming (coding) of commands is usually executed. This means, however, increased rigidity of the application and poor maintainability.

When the chip cards are installed, an encryption of the data or the commands for authentication of the chip card or of the world external to the chip card is generally used. To do this, a symmetrical encryption algorithm such as DES (digital encryption standard) or an asymmetrical encryption

algorithm such as the public key algorithm is generally used. A Personal Identification Number (PIN) is used almost exclusively for authentication of a chip card holder at this time. In addition to devices such as read/write devices for chip cards, there are also especially further elements, such as authentication elements for encryption and for input of the PINs, which must be taken into consideration for an application with chip cards and which may be involved in such an application.

Communication of the elements and devices, which is realized by hardware or software with the external application segments, is performed through the use of programming interfaces. For a chip card, however, a simple interface is not satisfactory. Due to the complexity of the data structures and also of the required commands, it is not enough to use the imbedding of the commands in another protocol or the use of a higher programming language. For application specific data, this also presumes knowledge of the internal structures of the chip card and hence of their accessibility and their method of access through the application via the interface to the chip card.

SUMMARY OF THE INVENTION

The object of the invention is to create a simplification of the interface for communication with a chip card. This object is achieved in accordance with the independent claims.

An apparatus according to the invention for communication of an interface with a chip card usually has at least one application data storing means (also known as the dictionary) for recording of information concerning application specific data of the application, and at least one chip card dialog module (also known as the agent) for the generation of commands using the application data storing means for an interface to the chip card, whereby the chip card dialog module contains card specific data concerning the chip card.

A request for communication with the chip card is made by the application to the chip card dialog module specified for the chip card. In response to the request, the chip card dialog module generates at least one command which is required for communication with the chip card. To achieve this, the chip card dialog module employs application specific information which is stored in the application data storing means.

In order to facilitate the implementation of different types of chip cards, it is necessary to obtain an interface of the application to the chip card which is as uniform as possible. In accordance with the invention, there is a separation between application specific data and card specific data. Application specific data are those data which contain information concerning the type, location, size and access methods for the data stored on a chip card, as well as the actual data stored on the chip card. On the other hand, card specific data represent those data which provide information concerning the commands and the protocol of a chip card which are required for access to the data stored there.

The separation between application specific data and card specific data makes it possible that one and the same application may be implemented with different types of chip cards. This leads to a significant simplification of the interface to the chip cards, and also improves the separation between internal application segments on the chip card and application segments which are external and separated from the chip card. In this manner, a flexible adaptation to new applications and chip card types is supported.

The solution described here facilitates the integration of chip cards in existing applications and also the implementation of new applications. By the separation of application specific and card specific data, the knowledge required for applications is minimized, and their use is simplified. The maintainability of the internal as well as external application segments is significantly improved. The application of the chip card is relieved of card specific aspects such as commands, protocols and data structures. At the same time, an asynchronous operational mode is made possible, in addition to the synchronous mode.

The invention has preferential applications in read/write devices for chip cards in the broadest sense, that is, in PC·s or other devices which coordinate, control and either directly or indirectly perform communication with chip cards.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. **1** a schematic representation of the communication of an application with a chip card in accordance with the invention,

FIG. **2** the use of several dictionaries and/or several agents for a number of different applications and types of chip cards,

FIG. **3** a further embodiment which makes possible a reduction of overhead in the communication with the chip card.

DESCRIPTION OF THE PREFERRED EMBODIMENT

FIG. **1** depicts a schematic representation of the communication of an application **10** with a chip card **20** in accordance with the invention. The application **10** is separated from chip card **20** and may be located, as an example, in a read/write device, a computer, or any other environment which is capable of communication with a chip card **20**. An application data storing means, the so called application dictionary **30**, serves for the recording of application specific data. A chip card dialog module, the so called agent **40** (smart card agent or smart card interpreter), generates the commands required for the interface of the chip card **20**.

The dictionary **30** contains information concerning the type, location, size and access methods of the data stored on the chip card, as well as information for the processing of these data which may be necessary based on security options of the application **10**. For simple identification of the data concerned for access through an application **10**, the data are assigned one or more alias names. This information is filed in the dictionary **30** in an appropriate form, such as in a table or hierarchy, and contain all of the necessary information for the application **10**, or for a number of additional applications (for further details regarding this, refer to FIG. **2**). Access to the dictionary **30** takes place through the agent **40**. The assignment of a dictionary **30** to an agent **40** is effected through the external application segment or a special extension of the agent **40**.

The generation of the dictionary **30** is preferably brought about by the manual creation of the required data. To achieve this, all of the information for all of the data on the chip card **20**, including their characteristics, are necessary. The data on the chip card **20** which are accessible from the application **10** are provided with alias names which are intended for the dictionary **30**.

If the data of the chip card **20** are to be processed, the application **10** submits a request **100** to the agent **40**. A request **100** of this type contains information concerning the desired access method, as well as the alias names of the desired data. In response to this request **100**, the agent **40** generates the command sequence required in order to access the data on the chip card **20**. To achieve this, it makes use of the information which is stored in an entry for each alias name concerned in the dictionary **30**. The necessary parameters of the access method are given to the agent **40** with the request **100**. The alias name of the data, the read method to be employed, and data concerning the filing location of the answer to the request **100**, for example, belong to a request **100**, which is to perform a reading of a date on a chip card. This type of request may also consist of the access mode READ for data with the alias name ACCOUNT NUMBER, and a memory address for the filed data. In addition to the read and write methods, authentication, generation or deletion of structures, as well as the application specific commands which may be implemented with chip cards may also occur here. In these instances there are methods stored on the chip card **20** which are initiated by special commands. For example, this may be the modification of an amount in a cash account stored on the chip card **20**.

In order to make simple imbedding into existing systems possible, it is frequently required that an asynchronous functional mode of the agent **40** be ensured. Many of the software platforms in use require event controlled programming. A synchronously functioning module could possibly lead to a temporary blockage of the entire system. However, the duration of this blockage is frequently not supportable for these chip card applications, since this includes a data transfer to a relatively slow medium as well as the processing of commands of a relatively slow working processor.

The agent **40** is preferably implemented as an asynchronous module, but may also be operated synchronously. Upon the request **100** of the application **10**, the agent **40** generates a sequence of commands for the chip card. Each command **110** of the sequence is given back to the application individually. The application transfers the respective command **110** of the agent **40** as an application command **130** to a corresponding read/write device **120** for direct reading and writing of data on the chip card **20**. At this point it should be mentioned that the application **10**, as has been described above, may also be situated directly in a read/write device **120**. For certain applications, a single command **110** may be adequate instead of a series of commands.

The read/write device **120** transfers the application command **130**, or an application command **130'** derived from it, to the chip card **20** and receives from it the response **140**. The data of the response **140**, or those of a response **140'** similarly derived from it, are accepted by the application **10** and in turn fed to the agent **40** as response data **150**. The agent interprets the response data **150** and generates thereupon the next command if necessary. The process is repeated until the desired data have been processed. At the end of the process, or also continuously through the command sequence, the agent **40** transfers a data record **160** as a reaction to the request **100**, and the data returned from the chip card **20** to the application **10**. This data record represents the response of the chip card **20**, which is intelligible for the application **10**, to the request **100** of the application **10** which was not intelligible to the chip card **20**.

The implementation of the agent **40** is performed preferably through the use of the concept of the finite automatic machine. This type of automatic machine is capable of storing its own internal condition and, based on input data and its current condition, to convert to another condition. Programming of this automatic machine is performed through the pre-selection of the desired condition conver-

sions for the specific combinations of momentary condition and input data.

During generation of the commands, there are also instances of the use of data or commands which are protected by codes, with the resulting required encryption/decryption, for example on the chip card **20**. For reasons of security, these should generally not be performed by the agent **40** itself. The agent **40** will therefore, through an appropriate request to the application **10**, signal this requirement and deliver the required data. The application **10** will in most cases in turn pass this request on to a specialized module for encryption. After the response, the agent **40** will continue the generation of the command sequence.

In addition to requests for encryption, requests may also be made to the read/write device **120**, as well as requests to a PIN input device (not shown), among others. These requests are passed on by the application **10** to the appropriate devices for processing.

In order to communicate with the chip card **20**, the application **10** will first select the appropriate agent **40** (for example, from a number of possible agents). At the latest point, the desired dictionary **30** is also specified at the time of the generation of a request to an agent **40**. Both take place based on special response data of the card, which are received, for example, through an initial query from the chip card **20** and by knowledge of the specific application **10** which is performing the transaction. After the selection of the agent **40** and the dictionary **30**, the request **100** may be submitted to the agent **40** by the application **10**. At that point, the agent **40** searches the dictionary **30** for the alias names contained in the request **100** and receives then the necessary application specific information concerning them. The first request **100** places the agent **40** into a start mode and, upon this, the first command of the required command sequence is generated. With each response to a command, the agent **40** will alter its internal condition and gather information concerning the progress of the sequence. The generation of commands is concluded as soon as the agent **40** has reached an end condition which is relevant for the request. This is the case if an error should have occurred in communication, or if the data have been successfully processed.

If an alteration of the application **10** through a modification of the data stored on the chip cards should be necessary, then this may be limited to an alteration of the dictionary **30**. A new dictionary **30'** (not shown) is made available to the application **10**, and the application **10** itself does not need to be changed. An alteration of the segments of the application **10** relating to the chip card **20**, such as the use of chip card types with a different interface for example, leads to the generation of a new agent **40'** (not shown). The information related to the data of the application **10** in the dictionary **30** or **30'** do not have to be altered in this instance.

It is to be understood that the command **110** and the response **140** of the agent **40** are preferably not changed either through the application **10** or through the read/write device **120**, or any other possible devices located between the agent **40** and the chip card **20**. The signals **110**, **130** and **130'** are in this instance identical, and the chip card **20** receives the command **110** of the agent **40** indirectly and transfers to it in turn (again indirectly) its response **140** to the command **110**. Hence, the signals **140**, **140'** and **150** are also identical. In a similar manner, the direct transfer of the command **110** to the chip card **20**, and of the corresponding response **140** back to the agent **40**, may be attained in each instance. It is also to be understood that the agent **40** may be operated in an asynchronous as well as a synchronous mode. In the asynchronous mode, the application **10** does not wait for a reply from the agent **40** to the request **100**, and is hence also available between the request **100** and its reply. In the synchronous mode, however, the application **10** waits for a reply from the agent **40** to the request **100**, and is hence not available until after the completed reply.

FIG. **2** depicts the use of several dictionaries **230** and/or several agents **240** for a number of differing applications **210** and types of chip cards **220**. A so called agency **250** is invoked by the application **10** from the number of applications **210** instead of the agent **40**, as in FIG. **1**. It is the task of the agency **250** to manage the various dictionaries **230** and agents **240**, and to provide for the availability of corresponding versions as required. To achieve this, a list of all local dictionaries **230** and agents **240**, with their characteristics and also their criteria, is preferably used when these are to be applied. If there is no appropriate agent and no appropriate dictionary available, the necessary original or a copy thereof may be requested from a different agency **260**. This may be performed through a local network as well as over appropriate communication paths between the agencies **250** and **260**.

If, in the implementation of an agent **40**, an interpreted program language is applied, one and the same implementation may also be applied on devices with differing hardware and operating systems. To achieve this, an appropriate interpreter is implemented for each device, which processes the program of the agent through interpretation.

If, during the transport of the commands from the agent **40** to the chip card **20**, several program levels are to be traversed, significant overhead may then result in the communication with the chip card **20**. This may be reduced if, as is depicted in FIG. **3**, the agent **40** is placed closer to the chip card **20**. To achieve this, the agent **40** is replaced by a substitute, the so-called proxy **340**. The proxy creates a single data record **350** for each request **100** with all of the required information from the dictionary **30** and allows this to be sent to a router **360**. The router **360** receives the data record **350** of the proxy **340**, and then in turn instructs the agent **40**, which has been shifted to this level. All of the commands **370** generated by the agent **40** are now sent to the chip card **20** through the router **360**, and every response **380** is sent in return to the agent **40**. A result **390** of the request **100** after communication of the agent **40** with the chip card **20** is finally given back to the proxy **340** by the router **360**, and then passed on by the router as the response **160** to the application **10**. The proxy **340** ensures a uniform interface for the application **10**, so that it does not have to be modified in its use. The application **10** remains independent of whether it communicates with the agent **40** or its proxy **340**. The router **360** is preferably placed on the same level as the agent **40**, while the dictionary **30** remains on the level of the application **10** or of the agency **230**.

It should be noted that the read/write device **120** exemplifies an abstract representation of a device with a direct access to the chip card **20**. The read/write device **120** may nevertheless consist of a number of individual devices and layers with individual interfaces among each other.

It is to be understood that the invention relates to simple memory boards as well as smart data carrier cards with their own processor for the control of the data stored on the chip of the data carrier card.

A modification of the dictionary **30** may also be effected in accordance with the invention directly through the agent **40** or the agency **250**. This is especially an advantage when structures on the chip card **20** are to be created or modified

using the agent **40**, for example when the agent **40** is to store or restructure a new file on the chip card **20**.

In one embodiment of the invention, the dictionary **30** is located directly on the chip card **20**. Through this, it is possible that the necessary dictionary **30** is always directly available to the chip card **20**. In a further embodiment, the dictionary **30**, which is located directly on the chip card **20**, is copied by this chip card **20** and stored (preferably by the agent **40**), and is available to the agent **40** or the agency **250** for further chip cards of the same chip card type.

Operation of the Preferred Embodiment

The following example is intended to show the process flow of a command sequence. The application **10** reads an account number from chip card **20**. The agent **40** to be used is selected based on the card characteristics of the chip card type of the chip card **20**. The application **10** sends the request **100** to the agent **40**, to which a specific dictionary **30** has already been assigned by a previous selection, based on the card data of the chip card type of the chip card **20**. Table **1** depicts an excerpt from this assigned dictionary **30**.

The request **100** contains as parameters the command "READ", the alias name "ACCOUNT NUMBER", and the memory address of a buffer in which the result of the request **100** is to be filed. The agent **40** then searches the dictionary **30** for the alias name "ACCOUNT NUMBER" (refer to Table **1**). When found, the entry provides the information as to the directory and the file of the chip card **20** in which this data item is to be found, as well as information regarding the type and size of the data item, and security relevant information. The data on the chip card **20** in this example are arranged in directories such as are typically understood for data systems of computers.

In this example, the path indicator "3F00.A100.4001" is found in Table **1** (marked with an arrow) under the alias name "ACCOUNT NUMBER", and also the information that the type of the file is TRANSPARENT. The data item is located in the file with the offset **4** and a length of 5 bytes. The indication of the code domain CREDIT and the authorization domain AUT_CREDIT serve for translation of the logical code numbers into physical numbers. Upon entry of the passwords (with a PIN) the domain GLOBAL is used for PIN1 and PIN2. A domain for a specific object represents a group of directories and files, which access a common entity of these objects.

The agent **40** initiates the generation of a command sequence for the selection of the required directory on the chip card **20** based on the path indicator "3F00.A100.4001". In this, the current directory selection is taken into consideration. In the event that the directory 3F00 has already been selected, the first command **110** which is generated will be for selection of the subdirectory A100. This first command **110** is passed back to the application **10** with a signal that the command sequence should be continued. The application **10** then forwards the first command **110** as a command **130** to an interface for transmitting to the read/write device being used by the chip card **20**. The device then transmits the command **130** to the chip card **20**. The chip card **20** sends a response **140** with a confirmation of the selection and, if necessary, other information about the selected directory. The response **140** is returned by the read/write device **120** to the application **10**, which immediately passes it on as response data **150** to the agent **40**.

The response data **150** are presented in general in a format which is not intelligible for the application **10**. In order to be able to understand the response data, extensive knowledge of the agent **40** must be available through the chip card **20**. The agent **40** is able to form conclusions from the response data **150** as to the success of the first command **110**, and to gather information regarding the progress of the command sequence. As a next step, a second command **110'** for the selection of the file 4001 is communicated to the application **10**, which is treated as was the first command **110**.

Finally, an additional command **110"** is generated for reading of the requested data. For this, the information as to the location (Offset=4) and size (length=5) of the account number is used. The command **110"** is transmitted as the command **110** previously was, and correspondingly answered by the chip card **20**. The response data **150"** of the chip card **20** also contain at this point the desired data. These data may be as an example encrypted, which the agent **40** recognized based on previous information during the selection of the file. At this point, a request **160** of the agent **40** will be generated to the application **10** in order to perform the encryption of the data. The application **10** may itself perform the encryption, or it may send it on to an additional specialized module. The result is returned as the response data item **150'"** to the agent **40**. The agent now copies the data in a format which is intelligible to the application **10** into the buffer memory which was indicated upon the original request **100** and signals the end of the command sequence to the application **10**. The application **10** may now access the account number and send the next request to the agent **40**.

What is claimed is:

1. Apparatus for responding to an application request by generating a command sequence to access a chip card, comprising:

reading means for reading card type information from the chip card;

an external layer separate from the chip card comprising:

one chip card dialog module selected from a plurality of chip card dialog modules using the card type information, the chip card dialog module containing card specific means responsive to application data information for generating commands in accordance with a protocol of the chip card;

at least one application data storing means separate from the dialog module for storing the application data information pertaining to type, location, size and access methods of data stored on the chip card;

more than one of the plurality of chip card dialog modules using the same application data information for generating commands to interface to respective chip cards;

whereby the application is relieved of commands protocols of a plurality chip cards that may be used with the application.

2. Apparatus of claim **1**, further comprising:

an allocation device for allocating an application data storing means to a chip card dialog module.

3. Apparatus of claim **1**, wherein the information for one or more applications is contained in the application data storing means.

4. Apparatus of claim **1**, wherein the chip card dialog module is implemented as a finite automatic machine.

5. Apparatus of claim **1**, further comprising:

at least one agency which is responsible for administration of each type of chip card dialog module and application data storing means, and which is used by the application to determine the appropriate combination of chip card dialog module and application data storing means for each type of chip card.

6. Apparatus of claim **5**, wherein at least one agency is linked with another agency through a communication path.

7. Apparatus of claim 1, further comprising:

a proxy which is placed substantially at the programming level of the application and which replaces the chip card dialog module in its communication with the application, the proxy providing a means for the creation of a data record, wherein the data record contains information from the application data storing means in response to a request of the application;

a router having a means for receiving the data record and a copy of the chip card dialog module;

wherein the router sends a command generated by the copy of chip card dialog module to the chip card, and the copy of the chip card dialog module returns a response from the chip card to the proxy and the proxy passes the response to the application.

8. A method for responding to an application request by generating a command sequence to access a chip card, comprising the steps:

receiving identification of a chip card:

applying to a chip card dialog module which is specified for the chip card from a plurality of chip card dialog modules, for card specific command generating steps in accordance with a protocol of the chip card, and

generating at the chip card dialog module, at least one command which is required for communication with the chip card, in accordance with the protocol of the chip card by accessing application specific information which is stored in memory in an application data storing means and wherein others of the plurality of chin card dialog modules may also access the same application specific information allowing the application to be relieved of the commands and protocols of a plurality of chip cards that may be used with the application.

9. A method of claim 8, further comprising the step of receiving the special response data from the chip card in response to an initial query to the chip card.

10. A method of claim 8, wherein;

the chip card dialog module of the requesting step is selected by the application from a number of chip card dialog modules; and

the application data storing means of the generating step is specified as part of the request to the chip card dialog module;

whereby both steps are based upon the special response data from the chip card and information concerning the specific application being performed.

11. A method of claim 8, wherein a first request places the chip card dialog module into a start mode, for generating a first command of a command sequence, in which the chip card dialog module alters its internal condition with each response to a command and gathers information concerning the progress of the sequence, and aborts the generation of commands as soon as the chip card dialog module has reached a relevant end condition for the request.

12. A method of claim 11, wherein the end condition is reached when an error occurs during the communication, and when data has been successfully processed.

13. A method of claim 8, further comprising the step of:

altering a specific application data storing means to create a new application data storing means for modifying the data stored on the chip card, without altering the application program itself.

14. A method of claim 8, further comprising the steps of:

placing a proxy substantially at the level of the application to replace the chip card dialog module in its communication with the application, and which, upon the request of the application, creates a data record which contains information from the application data storing means;

receiving at a router, a copy of the chip card dialog module and the data record;

sending a command from the router which was generated by the copy of the chip card dialog module to the chip card; and

receiving a response from the chip card at the copy of the chip card dialog module;

the router returning the response from the chip card to the proxy and the proxy passing the response to the application.

15. A method of claim 8, wherein an application data storing means which is located on the chip card is copied from the chip card and stored in external apparatus, making it available to the chip card dialog module.

16. Apparatus for responding to an application request by generating a command sequence to access a smart card, comprising:

reading means for reading card type information from the smart card; and

an external layer separate from the smart card comprising:

one smart card dialog module selected from a plurality of smart card dialog modules using the card type information, the smart card dialog module containing card specific means responsive to application data information for generating commands in accordance with a protocol of the smart card;

at least one application data storing means separate from the dialog module for storing the application data information pertaining to type, location, size and access methods of data stored on the smart card;

more than one of the plurality of smart card dialog modules using the same application data information for generating commands to interface to respective smart cards;

whereby the application is relieved of commands and protocols of a plurality smart cards that may be used with the application.

17. Apparatus of claim 16, comprising:

at least one agency which is responsible for administration of each type of smart card dialog module and application data storing means, and which is used by the application to determine an appropriate combination of smart card dialog module and application data storing means for each type of smart card.

18. Apparatus of claim 16, further comprising:

a proxy which is placed substantially at a programming level of the application and which replaces the smart card dialog module in its communication with the application, the proxy providing a means for the creation of a data record, wherein the data record contains information from the application data storing means in response to a request of the application;

a router having a means for receiving the data record and a copy of the smart card dialog module;

wherein the router sends a command generated by the copy of smart card dialog module to the smart card, and the copy of the smart card dialog module returns a response from the smart card to the proxy and the proxy passes the response to the application.

19. A method for responding to an application request by generating a command sequence to access a smart card, comprising the steps:

receiving identification of a smart card:

applying to a smart card dialog module which is specified for the smart card from a plurality of smart card dialog modules, for card specific command generating steps in accordance with a protocol of the smart card, and

generating at the smart card dialog module, at least one command which is required for communication with the smart card, in accordance with the Protocol of the smart card by accessing application specific information which is stored in memory in an application data storing means and wherein others of the plurality of chin card dialog modules may also access the same application specific information allowing the application to be relieved of the commands and protocols of a plurality of smart cards that may be used with the application.

20. A method of claim **19**, further comprising the step of receiving the special response data from the smart card in response to an initial query to the smart card.

21. A method of claim **19**, wherein;

the smart card dialog module of the requesting step is selected by the application from a number of smart card dialog modules; and

the application data storing means of the generating step is specified as part of the request to the smart card dialog module;

whereby both steps are based upon the special response data from the smart card and information concerning the specific application being performed.

22. A method of claim **19**, wherein a first request places the smart card dialog module into a start mode, for generating a first command of a command sequence, in which the smart card dialog module alters its internal condition with each response to a command and gathers information concerning the progress of the sequence, and aborts the generation of commands as soon as the smart card dialog module has reached a relevant end condition for the request.

23. A method of claim **22** wherein the end condition is reached when an error occurs during the communication, and when data has been successfully processed.

24. A method of claim **19**, further comprising the step of:

altering a specific application data storing means to create a new application data storing means for modifying the data stored on the smart card, without altering the application program itself.

25. A method of claim **19**, further comprising the steps of:

placing a proxy substantially at the level of the application to replace the smart card dialog module in its communication with the application, and which, upon the request of the application, creates a data record which contains information from the application data storing means;

receiving at a router, a copy of the smart card dialog module and the data record;

sending a command from the router which was generated by the copy of the smart card dialog module to the smart card; and

receiving a response from the smart card at the copy of the smart card dialog module;

the router returning the response from the smart card to the proxy and the proxy passing the response to the application.

26. A method of claim **19**, wherein an application data storing means which is located on the smart card is copied from the smart card and stored in external apparatus, making it available to the smart card dialog module.

* * * * *

US005689561A

United States Patent [19]

Pace

[11] Patent Number: 5,689,561

[45] Date of Patent: Nov. 18, 1997

[54] **COMPUTER-BASED TRADING CARD SYSTEM AND METHOD**

[76] Inventor: **Michael Pace**, EMA Multimedia, Inc., 11911 San Vicente Blvd., Suite 225, Brentwood, Calif. 90049

[21] Appl. No.: **540,209**

[22] Filed: **Oct. 6, 1995**

[51] **Int. Cl.**[6] **H04K 1/00**

[52] **U.S. Cl.** **380/4**; 395/186; 463/2; 463/29; 380/21

[58] **Field of Search** 380/4, 21; 395/186, 395/187.01; 364/410, 479.01; 463/1, 2, 29, 43

[56] **References Cited**

U.S. PATENT DOCUMENTS

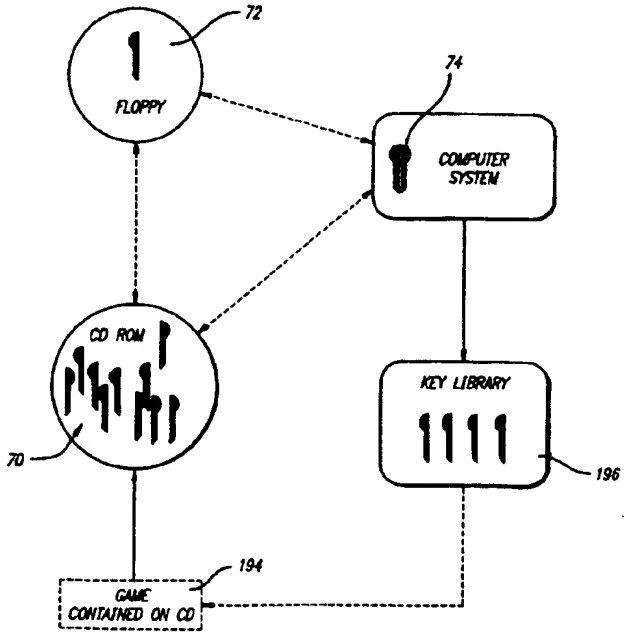
4,634,807	1/1987	Chorley et al.	380/4
4,905,280	2/1990	Wiedemer	380/16
5,109,413	4/1992	Comerford et al.	380/4
5,112,051	5/1992	Darling et al.	463/45
5,237,610	8/1993	Gammie et al.	380/10
5,273,288	12/1993	Teshima et al.	463/41
5,319,705	6/1994	Halter et al.	380/4
5,343,524	8/1994	Mu et al.	380/4
5,343,527	8/1994	Moore	380/4
5,379,433	1/1995	Yamagishi	395/186
5,411,258	5/1995	Wilson et al.	463/6
5,411,259	5/1995	Pearson et al.	463/36
5,412,717	5/1995	Fischer	380/4
5,412,718	5/1995	Narasimhalu et al.	380/4
5,416,840	5/1995	Cane et al.	380/4
5,437,464	8/1995	Terasima et al.	463/43
5,440,631	8/1995	Akiyama et al.	380/4
5,442,706	8/1995	Kung	380/30
5,533,124	7/1996	Smith et al.	380/4

Primary Examiner—Thomas H. Tarcza
Assistant Examiner—Pinchus M. Laufer
Attorney, Agent, or Firm—Oppenheimer Poms Smith

[57] **ABSTRACT**

A collection system using a CD ROM computer system wherein the collection items (or more specifically their unlocking keys) are contained in various floppy disks. With the disks inserted into the computer system, the icons of the collection items on the disks appear on the computer monitor. By (double) clicking on the icon, the unlocking key unlocks the corresponding collection item in the CD ROM program into the hard drive and at the same time the unlocking key is rendered inoperative. A generally reverse process can be used to lock the collection item relative to the hard drive and render the unlocking key operative. The user collects the collection items by unlocking, using a number of floppy disks, the locks in his CD ROM program, which contains the corresponding locks for all of the collection items in the set. When the entire set or a predetermined subset thereof has been collected, the CD ROM program allows the user to play an interactive game related to the collection items. The user can also enjoy a video and/or audio presentation contained in the disk and/or the CD ROM program associated with each of the keys (and thereby the corresponding collection items) by (single) clicking on the appropriate icons. That is, instead of the prior art system of trading paper cards or the like, the user herein trades floppy disks to collect the collection items, and can enjoy audio/visual presentations and interactive computer games also associated with the collected items.

29 Claims, 6 Drawing Sheets



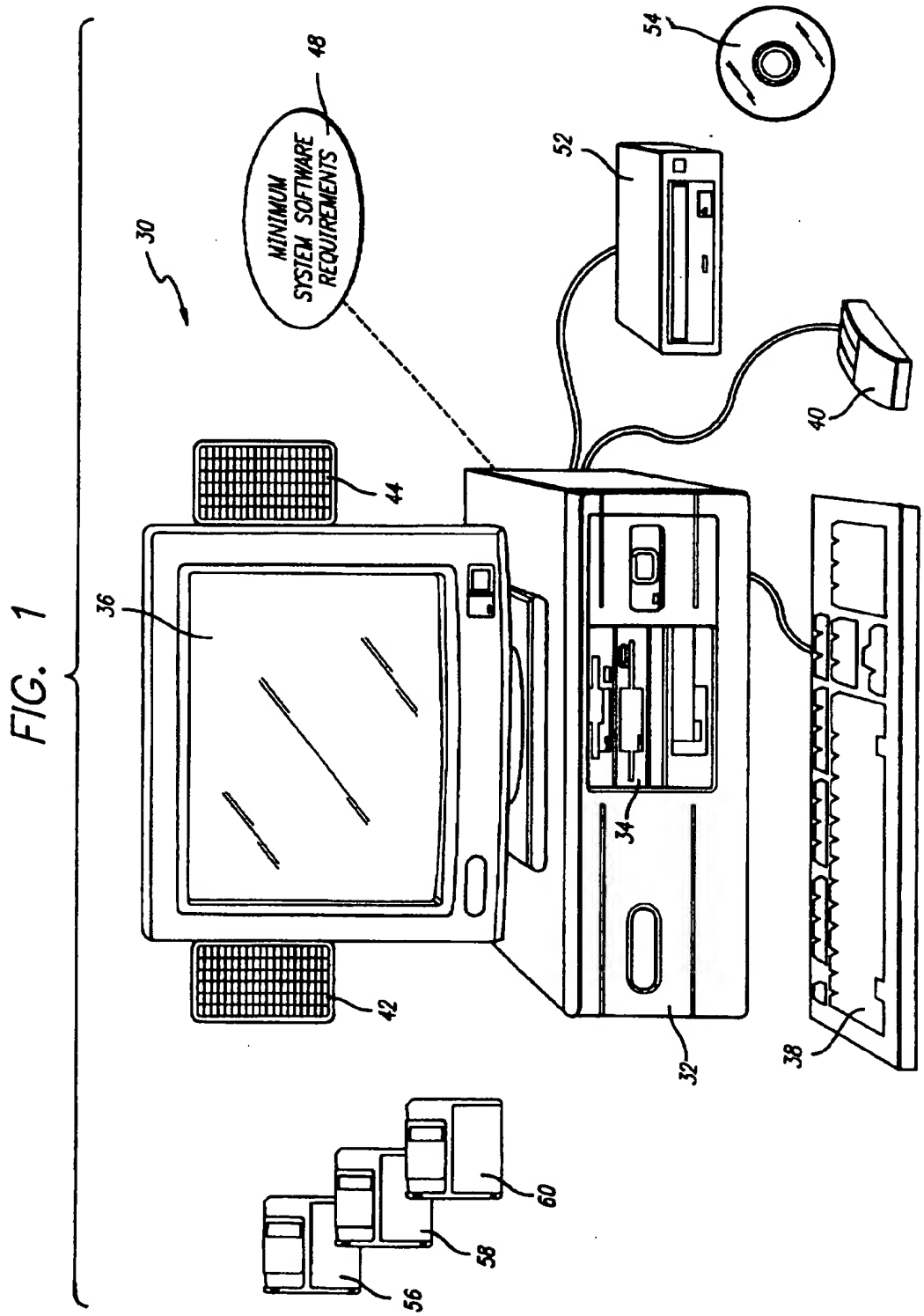

MINIMUM SYSTEM SOFTWARE REQUIREMENTS
30
32
34
36
38
40
42
44
48
52
54
56
58
60

FIG. 1

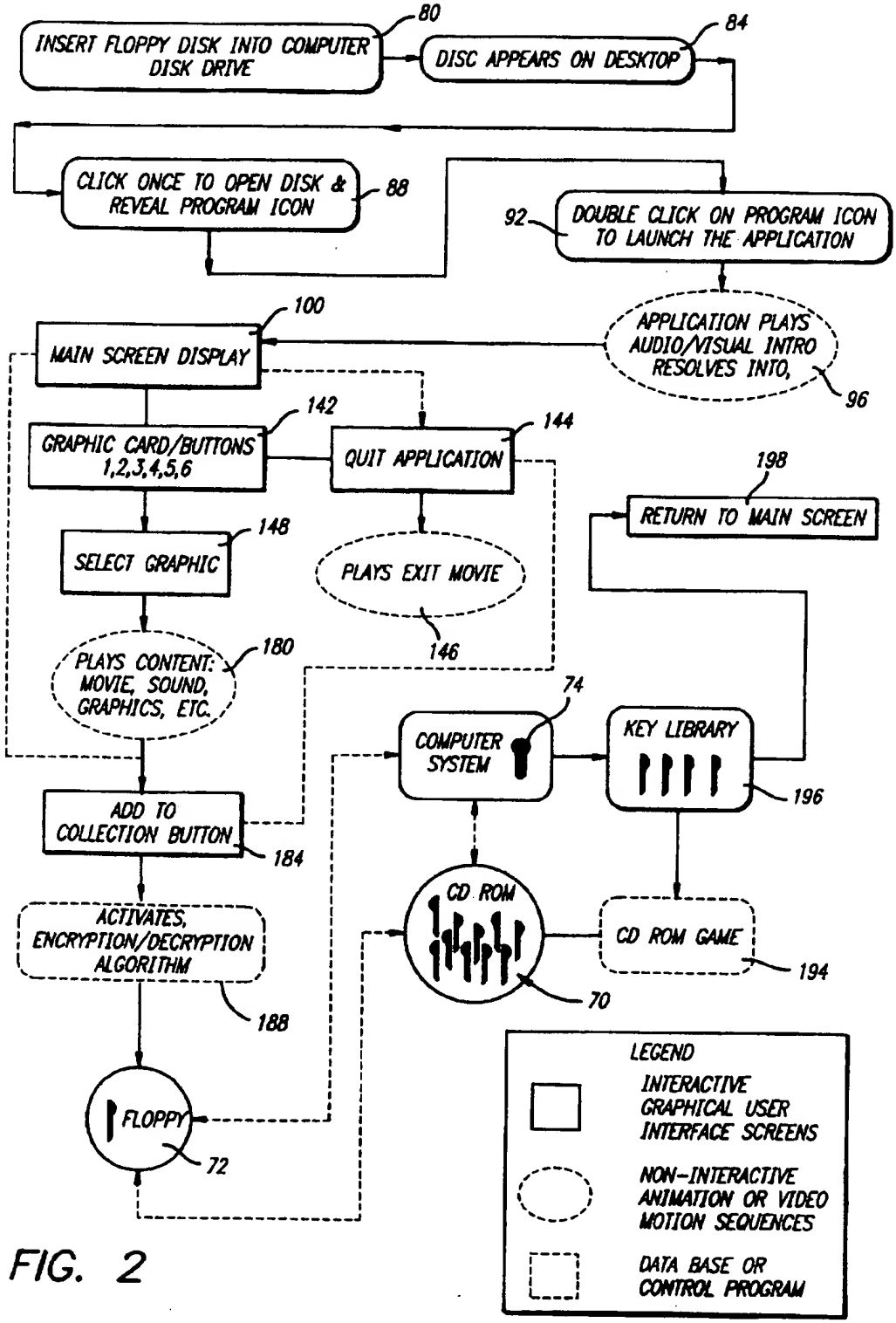

INSERT FLOPPY DISK INTO COMPUTER DISK DRIVE
80
DISC APPEARS ON DESKTOP
84
CLICK ONCE TO OPEN DISK & REVEAL PROGRAM ICON
88
92
DOUBLE CLICK ON PROGRAM ICON TO LAUNCH THE APPLICATION
APPLICATION PLAYS AUDIO/VISUAL INTRO RESOLVES INTO,
96
100
MAIN SCREEN DISPLAY
142
GRAPHIC CARD/BUTTONS 1,2,3,4,5,6
144
QUIT APPLICATION
198
RETURN TO MAIN SCREEN
148
SELECT GRAPHIC
PLAYS EXIT MOVIE
146
180
PLAYS CONTENT: MOVIE, SOUND, GRAPHICS, ETC.
74
COMPUTER SYSTEM
KEY LIBRARY
196
ADD TO COLLECTION BUTTON
184
CD ROM
ACTIVATES, ENCRYPTION/DECRYPTION ALGORITHM
188
CD ROM GAME
194
70
FLOPPY
72
LEGEND
INTERACTIVE GRAPHICAL USER INTERFACE SCREENS
NON-INTERACTIVE ANIMATION OR VIDEO MOTION SEQUENCES
DATA BASE OR CONTROL PROGRAM

FIG. 2

FIG. 3

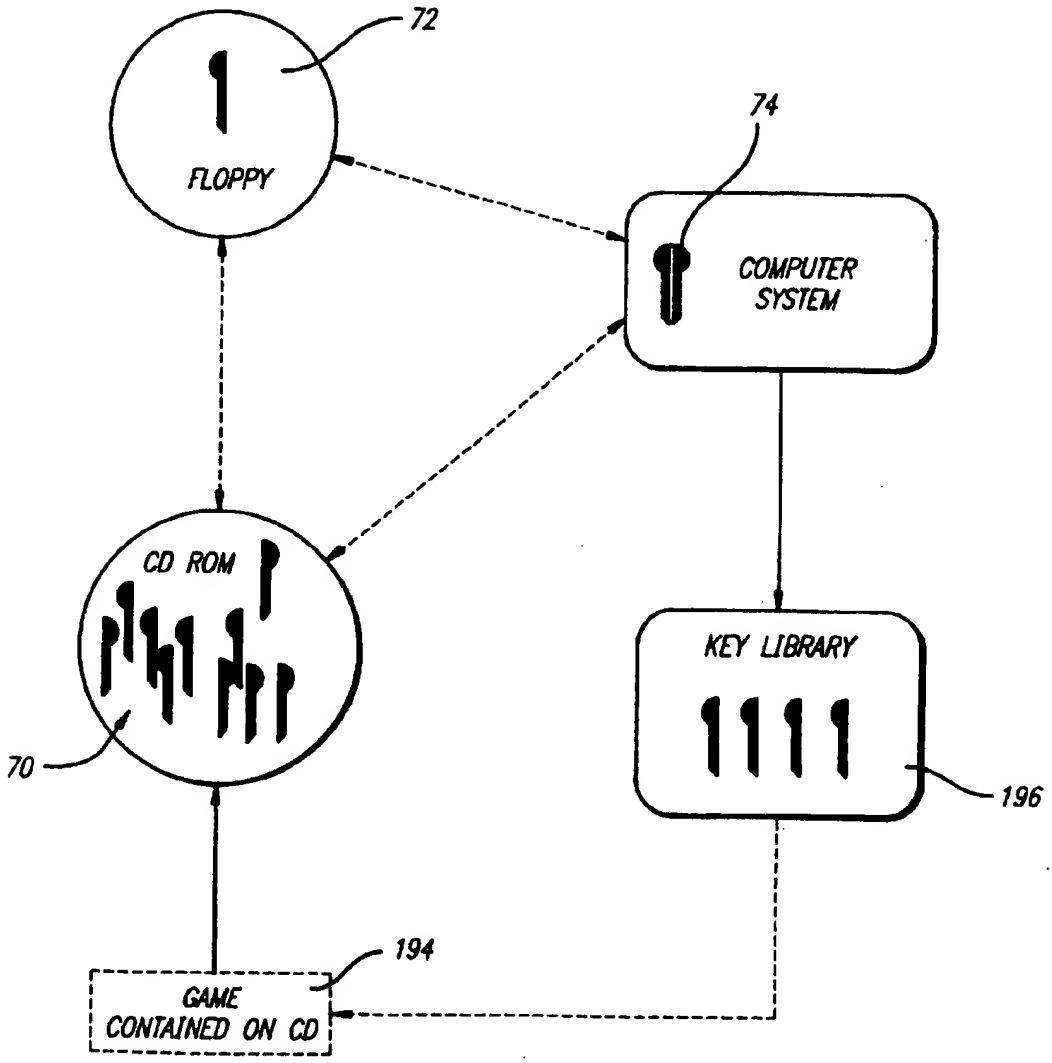

72
FLOPPY
74
COMPUTER SYSTEM
CD ROM
70
KEY LIBRARY
196
194
GAME CONTAINED ON CD

FIG. 7

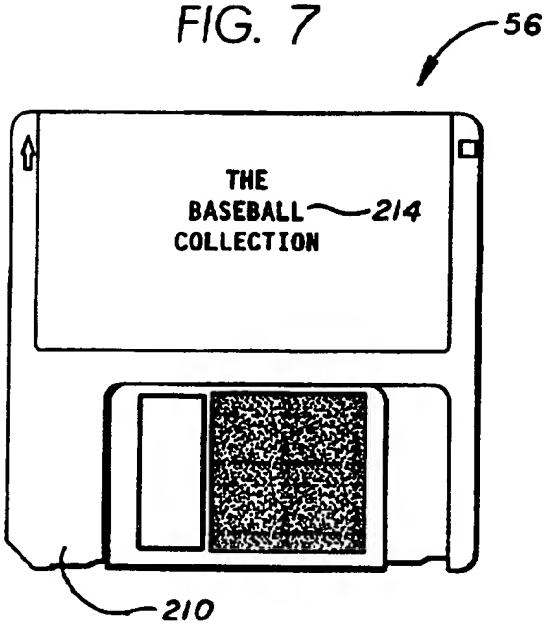


FIG. 8

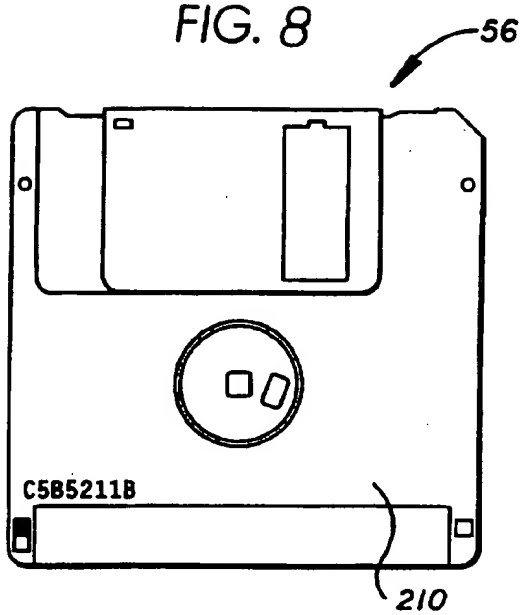


FIG. 4

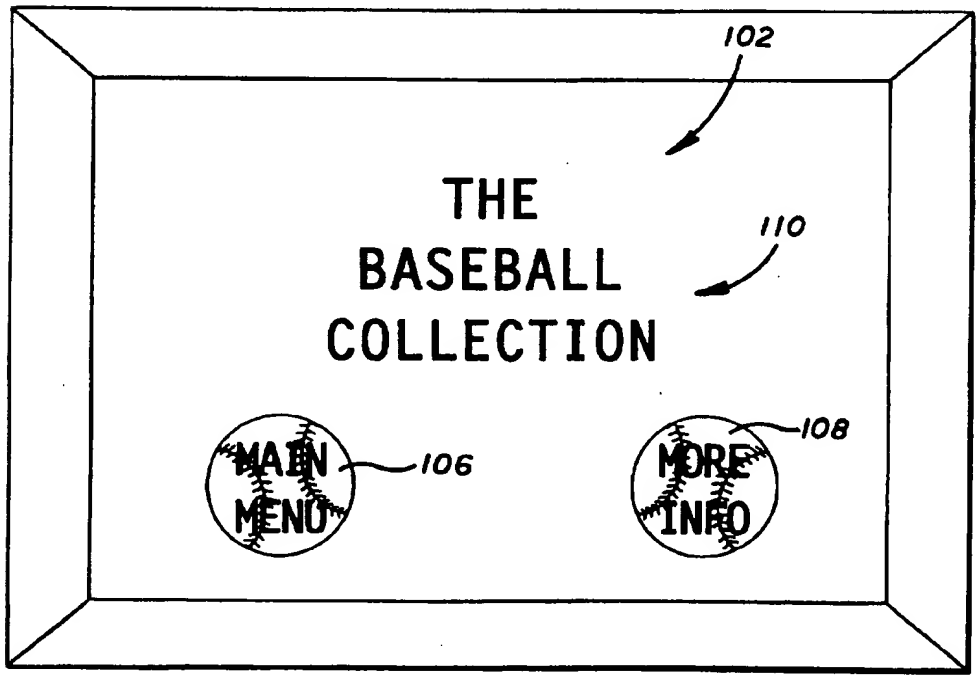


FIG. 5

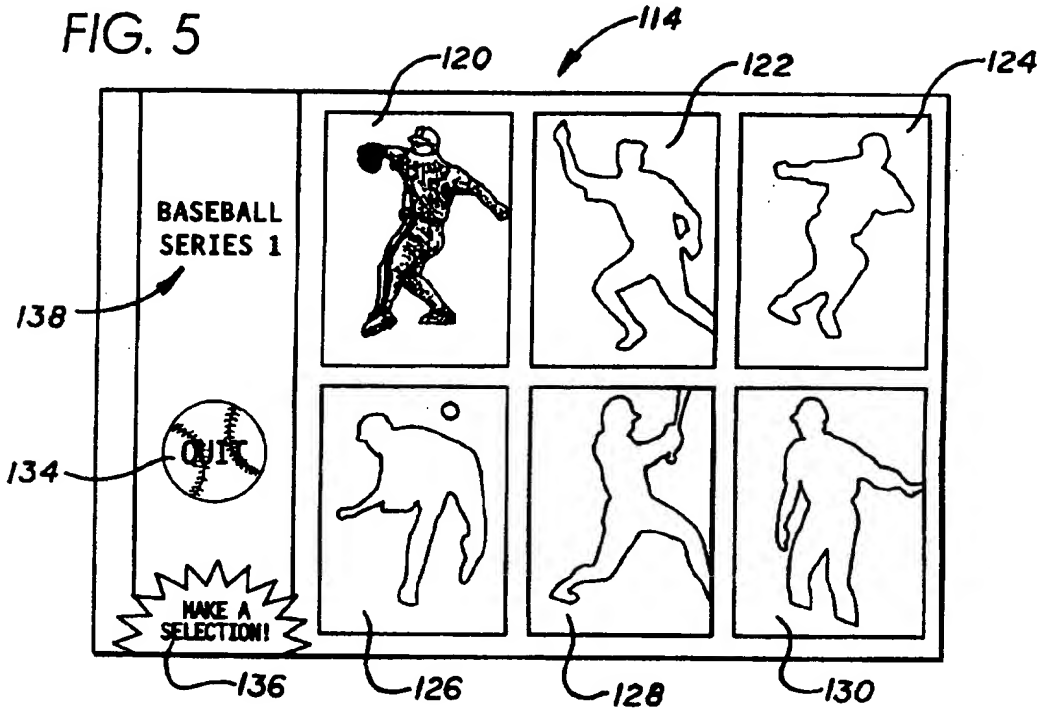

114
120
122
124
BASEBALL SERIES 1
138
QUIT
134
MAKE A SELECTION!
136
126
128
130

FIG. 6

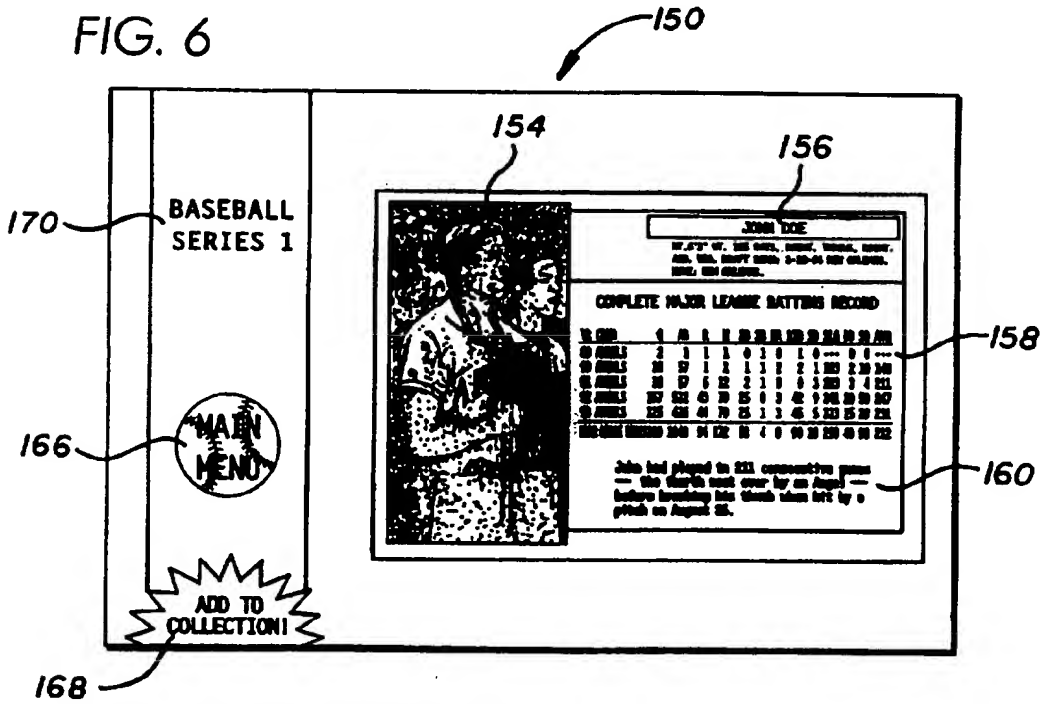
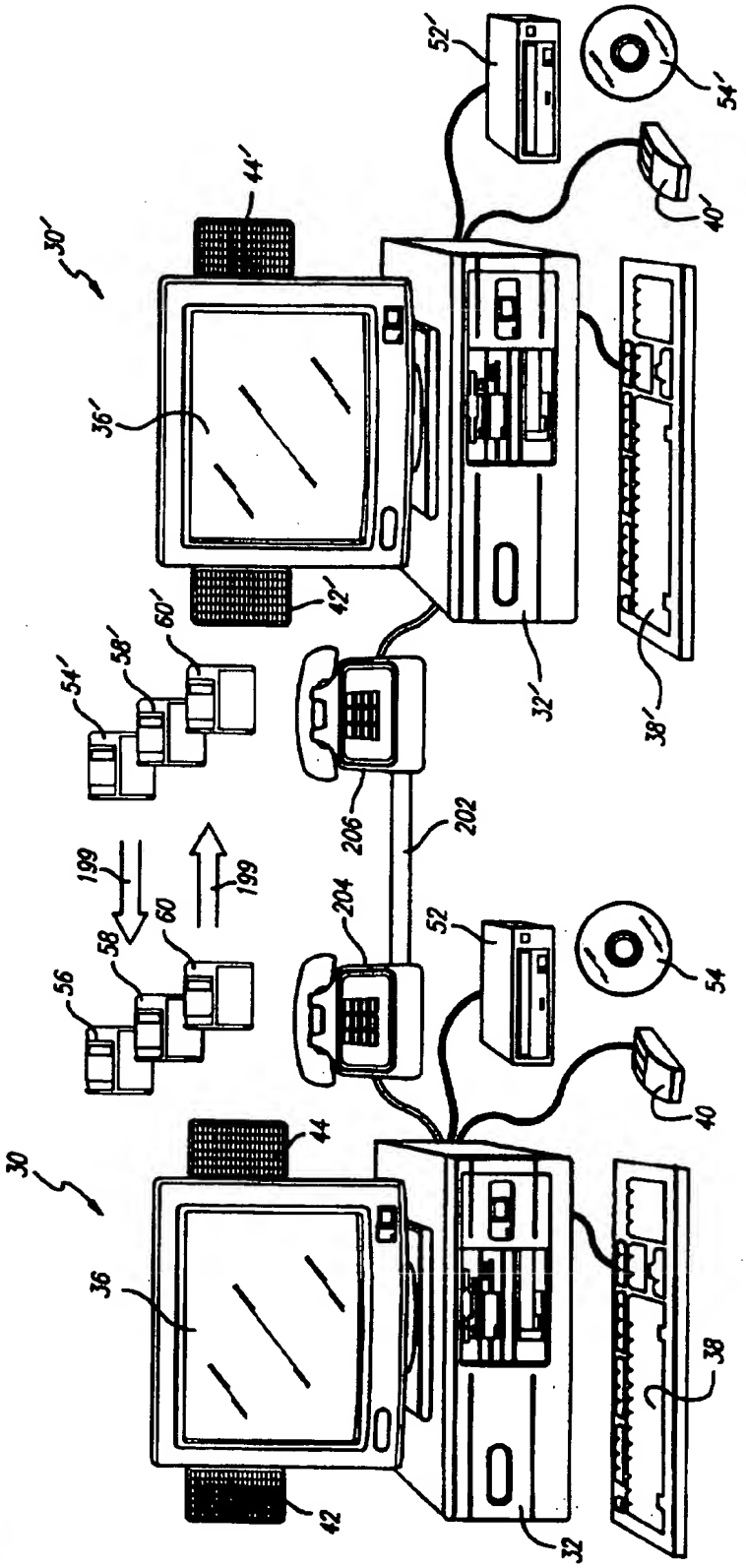

150
154
156
170
BASEBALL SERIES 1
JOHN DOE
COMPLETE MAJOR LEAGUE BATTING RECORD
158
MAIN MENU
166
160
ADD TO COLLECTION!
168

FIG. 9


52'
54'
40'
44'
30'
36'
42'
60'
58'
54'
32'
38'
206
202
199
199
204
60
58
56
52
54
40
44
30
36
42
32
38

COMPUTER-BASED TRADING CARD SYSTEM AND METHOD

BACKGROUND OF THE INVENTION

Through the years, people and especially children have enjoyed collecting various things as a game, a hobby, a competition, a lesson or for other reasons. In early times they collected stones, animal or plant parts or other things found in nature. Later it was also manufactured items such as marbles, toys, memorabilia, coins or stamps. More recently, collecting WACKY PACKS (cult stickers with humorous (sometimes ghoulish humor) corrupted pictures of common trademarked products) and POGS (a bottle cap type of game) have been popular. Over the past hundred or so years and continuing today, trading card collecting has been extremely popular and the more popular of them have been baseball trading cards sold in packs or in a package with another product typically bubble gum. The cards each have a baseball player pictured thereon with his playing statistics and a brief write-up on him. The collector, typically children, would seek to collect a desired set of cards, which could be all of the players on a certain team, his/her favorite players, or the more rare and desired and thus valuable cards. In fact, some cards are so desirable that they have sold for many tens of thousands of dollars.

To obtain the desired cards the collector can purchase them from the original source if still available and/or can trade his less desirable cards with other collectors for his desired cards. The negotiating and interactions with other collectors is a big attraction to the collection activity. While in the past and typically now the trading takes place informally with just two or a few people gathered together, large trading conventions attracting tens of thousands of people have been held to attempt to satisfy the growing trading interest, needs and demands. In other words, people are continually looking for different items to collect, and different more entertaining ways of trading and collecting, and this invention is directed to at least partially satisfying those needs.

SUMMARY OF THE INVENTION

Disclosed herein is a computer-based collection and trading system, providing a unique and flexible system for trading, displaying and enjoying collectible items. Instead of being contained on paper trading cards (or stamps or the like) as often found in the prior art, the collectible items of the present invention are contained on standard 3½" floppy disks, and the children (or adults) then trade the floppies themselves instead of the paper cards. To view the collectible items on a floppy, and each floppy would typically have one or more and preferably six items therein, the floppy is inserted into a compatible personal computer and the floppy program run. For each collectible item, an associated audio/visual display on the computer would be available to be accessed and enjoyed if desired.

The collectible items can be "transferred" from the floppies to a CD ROM program on that computer for further display, collecting and interactive game playing. The CD ROM program is like an album for the cards, providing a neat, organized and fun method of storing and retrieving the cards and accompanying software. The game playing is preferably possible only after the entire set of collectible items or one or more predetermined subsets thereof have been collected, using the CD ROM program into the computer's hard drive. This provides a greater incentive to collect the entire set.

The collectible items are individually "transferred" from the floppies to the hard drive using a system of encryption/decryption algorithms. When transferred, the item is disabled from that floppy so that the floppy cannot be used again to transfer that item to another computer system (unless the item has been transferred back to it from the hard drive). The item can be transferred from the hard drive only to that floppy (or a similar disabled floppy having that algorithm), as when the user wants to trade the item on the floppy to another collector. As each collectible key or code is located on a floppy, it is selected by the user, locked and dimmed on that floppy, its key code updated to a preferences file on the host computer system and then unlocked on the companion collection CD ROM program. That is, the key has been dimmed from use on the floppy and lit on the CD ROM via the computer system. All remaining keys remain intact on the floppy disk, available for trading to another collector. Once all the cards in a particular series are collected (unlocked from the CD ROM to the hard drive), an interactive game is decrypted for the collector-user to play and enjoy.

As further protection and deterrence against unauthorized copying, each floppy disk case includes affixed thereon a unique hologram and a unique serial number.

Other objects and advantages of the present invention will become more apparent to those persons having ordinary skill in the art to which the present invention pertains from the foregoing description taken in conjunction with the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a schematic diagram of equipment used in a computer-based collection system of the present invention;

FIG. 2 is a flow chart of a collection system and method of the present invention using the equipment of FIG. 1;

FIG. 3 is an enlarged view of the lower left portion of the flow chart of FIG. 2;

FIG. 4 is a first screen display of the present invention on the monitor shown in FIG. 1;

FIG. 5 is a second screen display;

FIG. 6 is a third screen display;

FIG. 7 is a front view of one of the floppy disks of the system of FIG. 1, shown enlarged and in isolation;

FIG. 8 is a rear view of the disk of FIG. 7; and

FIG. 9 is a schematic diagram of an alternative system of the present invention.

DETAILED DESCRIPTION OF PREFERRED EMBODIMENTS OF THE INVENTION

The computer equipment of the system shown generally at 30 are best illustrated in FIG. 1. The equipment includes a computer hard drive 32, a floppy disc drive 34, a display monitor 36, a keyboard 38, a mouse 40 and speakers 42, 44. The minimum equipment specifications and system software requirements 48 are MPC Level 2 or higher or MAC 68040 or higher. The requirements are itemized in detail later in this disclosure. Also, periodically throughout this disclosure prior art patents are referenced, and each of their entire disclosures is incorporated herein by reference.

A CD ROM program is installed in the CD ROM drive 52 using a CD ROM disk 54 in a conventional manner. The collection items and associated programs to interact with the CD ROM program are played and installed into the hard drive 32 by using floppy disks inserted individually into the

disk drive. More than one floppy disk is preferably needed for this system and three are pictured in FIG. 1 at 56, 58, 60.

The CD ROM program includes all of the collectible items in the set of collectible items. However, they are all "locked" and cannot be accessed and unlocked into the hard drive 32 unless the unlocking "keys" are accessed and used. A separate key is provided for each of the locked items, and the keys are provided on the floppy disks 56, 58, 60. The number of items in the set of collectible items will be at least two and preferably and more likely many tens or hundreds, depending on the CD ROM program. Each floppy disk 56, 58, 60 will have at least one key and more likely a plurality such as six keys.

Although there can be floppies having identical keys, most floppies will have different groups of keys. For example, floppy 56 can have keys A, B, C, D, E and F (for corresponding collection items), floppy 58 can have keys D, E, F, G, H and I, and floppy 60 can have keys B, C, D, G, H and I. In this brief example, key A appearing on only one floppy may be the most valuable and thus most tradeable and key D appearing on all three floppies is the least valuable. The CD ROM program game (to be discussed later) may make certain keys more valuable even if more common due to the use of the corresponding item in the game. Also, certain keys may be more valuable because of the desirability of the associated item.

The "keys" on the floppy discs 56, 58, 60 unlock the "locks" of the CD ROM program and individually unlock the collection items onto the hard drive 32 using dualencryption code technology. More particularly, the CD ROM program contains the complete set of encryption/decryption algorithms or "private keys" as shown in FIGS. 2 and 3 generally at 70 and the floppy contains a complementary set of encryption/decryption algorithms or "public keys", as shown generally at 72. (See U.S. Pat. Nos. 5,442,706 to Kung and 5,343,527 to Moore). The public and private keys 70, 72 are combined shown generally at 74 via standard multimedia-computer equipment as depicted in FIG. 1 to unlock the private media and lock the public media. This is accomplished via an encryption/decryption algorithm which allows the user to effectively unlock media on the CD ROM drive 52 and lock the media on the floppy disks 56, 58, 60.

The system (30) is operated (or the game is played) referring to FIG. 2 as follows:

(1) the CD ROM program is loaded onto the CD ROM by inserting the CD ROM disk 54 into the CD ROM drive 52 in a conventional manner; at this time all of the collection items at this time are locked relative to the hard drive 32;

(2) the user acquires a (3½") floppy disk 56 having a floppy disk program including at least one of the coded keys 72 (for unlocking the corresponding collection item(s)) and inserts it into the disk drive 34 as shown in the flow chart by step 80, and the disk appears on the desktop of the monitor 36, as shown by step 84;

(3) using the mouse 40 the user clicks twice to open the disk and reveal the program icon, as shown by step 88, and double clicks on the program icon to launch the application, as shown by step 92;

(4) alternatively, pursuant to another method, the floppy disk 56 instantly mounts and launches into an audio (via the speakers 42, 44) and visual (via the display monitor 36) opening presentation beginning with opening credits and resolving into an opening main screen display, as shown by step 100; the main screen 102 has "Main Menu" and "More Info" buttons 106, 108 which are depicted in FIG. 4;

(5) clicking on the "More Info" button 108 reveals a screen (not separately shown in the drawings) with explanatory information, and alternatively clicking on the "Main Menu" button 106 produces the graphical icon display, as Shown in FIG. 5 generally at 114, and having the six graphical icons 120, 122, 124, 126, 128, 130 (the six trading cards), a "Quit" button 134, a flashing "Make a Selection" message 136 and a title 138; this is shown by step 142 in FIG. 2;

(6) clicking on the Quit button 134, quits the application as shown by step 144 and results in an exit audio/visual display as shown by step 146;

(7) instead of step (6), clicking on the desired icon (trading card), as shown by step 148, produces the display 150 of FIG. 6, which is an enlarged view of the back side of that card with another picture 154 of the player, his name 156, his statistics 158 and a brief bio 160; the display 150 also includes a "Main Menu" button 166 (which when clicked on takes the user back to display 114 of FIG. 5), an "Add to Collection" button 168 and a title 170; the appearance of display 150 can be accompanied with an audio and/or audio/visual display as shown by step 180;

(8) clicking on the "Add to Collection" button 168, as shown by step 184, activates the unique code (or encryption/decryption algorithm) for that graphical icon (or trading card or collection item) as shown by step 188, and unlocks the first collection item from the CD ROM into the hard drive 32; in other words, before clicking on the "Add to Collection" button 168 the CD ROM program had not been accessed or even need be present; rather, the user can enjoy the audio and visual presentations of the screens 114 and 150 of FIGS. 5 and 6 simply through the programs on the floppies 56 (58, 60) when loaded into the computer;

(9) as the first collection item (120) is unlocked from the CD ROM into the hard drive 32, the first coded key 74 on the floppy 56 is rendered inoperative; that is, it is incapable of unlocking a corresponding first collection item in a similar CD ROM program; in the display 114 of FIG. 5 it is dimmed, as shown by the other five graphical icons;

(10) after step (9), by clicking on the dimmed icon button, the first collection item can be "transferred" back to the first floppy 56, as for example when the user wants to trade (or sell) the first collection item (or more specifically the first coded key, etc.) with another person; when it is transferred back, the first collection item is (re)locked relative to the hard drive 32 and the first coded key (74) is rendered operative, using the encryption/decryption code, and goes from dim to bright as in the display of FIG. 5; instead of transferring it back to that specific first floppy 56, it can be transferred to another floppy, such as 58 or 60, which has a similar inoperative first coded key; all subsequent steps outlined below will presume that the first collection item has not been relocked;

(11) other collection items are unlocked as desired using other corresponding coded keys on the first floppy disk 56 and the first floppy disk removed from the disk drive 34;

(12) second, third and so forth floppy disks (58, 60) are inserted in the disk drive 34, and further collection items unlocked using the coded keys thereon, as available and desired;

(13) the CD ROM program includes an interactive game 194 (see U.S. Pat. No. 5,411,258 to Wilson et al.),

which can only be played after all or one or more specific subsets of the collection items have been unlocked as shown by the key library **196**; that is and referring to FIGS. **2** and **3**, all of the library keys **196** have been collected; to access this game, the CD ROM has its own interface which when presently mounted on the computer system shows a tally of the keys collected and the remaining keys to be collected before the game is activated; the user double clicks on the CD ROM icon or it mounts instantly, revealing a main screen which shows the tally and remaining keys; when the remaining keys have been collected an additional interface becomes active which allows access to the supporting interactive game within the collection series; the program can contain a number of games accessed when different collections, subsets or libraries of keys **196** have been collected or collection items unlocked;

(14) after playing the game **194**, the user clicks on a quit button of that screen to return to the main screen **102**, as shown by step **198**, or return to their collection; and

(15) thereafter, other collection items can be unlocked or transferred back as described above, other games played as the program permits, and/or collection items (or their coded keys) traded with others by trading the floppy disks containing them (as shown in FIG. **9**) by arrows **199**.

In other words, when a disk **56** is inserted into the floppy disk drive **34**, the information contained on the disk can be viewed but the software locked thereto cannot be removed or copied. The user must click (step **146**) on the chosen cards (graphical icons) (**120**) to send their unique unlock code/key **74** to the companion CD ROM **72**. That card is then dimmed (locked or rendered inoperative) on the floppy **56** as shown in FIG. **5** and lit (unlocked) on the CD ROM. A preferences file is created and housed on the hard drive **32**, hidden within the system folder. This file tells the CD ROM which keys have been collected and which icons are effectively unlocked.

Referring to FIG. **3**, the encryption/decryption system controls access to media assets. By controlling access, a game can be made of the acquisition and trading of access keys and the access to the media files. By collecting keys **196**, media is unlocked on the system. Possibilities are available for having nested keys that once having been gathered can provide a composite key that will allow access to alternate media files or collector bonuses and allow for multi-stage collectible experiences.

More specifically, access keys are stored in two portions—one on a master media element, such as a CD ROM **72**, and the other on copy-protected floppy diskettes **56**. Both keys are required to unlock a media file **74**. A lock may be moved from the floppy disk **56** to a table on a hard disk drive **32**. The process physically removes the key portion from the floppy diskette **56** and places that encrypted key into the table **196**, which is normally filled with encrypted dummy keys. The actual keys are hashed into the table using a proprietary algorithm. Also part of the key is based upon a defect table held in the FAT (File Allocation Table) of the floppy **56**. This defect is required in order to transfer the key, which is stored in the bad section identified by the floppy defect. As an added protection, the LBE (Logical Block Entry) of the bad sector of the floppy **56** also is part of the key. Once the CD ROM public key **72** is matched with the floppy private key **74**, the media asset becomes active and accessible.

Once keys are placed in the table they allow decryption of the media using PGP (pretty good protection) or a similar dual drive **34** key encryption system. The keys **72** on the CD ROM **72** work with a number of similar keys **74** placed on the floppies. In this way each floppy **56**, even for the same media file, will have a unique key. The keys may be transferred to the original floppies **56** and removed from the hard disk **32**. This process only allows original floppies to be used for removal of codes as the key is specific to that floppy.

Various prior art encryption systems for different uses are disclosed in U.S. Pat. Nos. 5,237,610 (Gammie et al.), 5,319,705 (Halter et al.), 5,343,524 (Mu et al.), 5,379,433 (Yamagishi), 5,412,717 (Fischer), 5,412,718 (Narasemhalu et al.), 5,416,850 (Cane et al.) and 5,440,631 (Akiyama et al.).

A description of a preferred baseball interactive game **194** and its various steps and options follow. Each Baseball Series disk contains statistical information that is unique to the baseball personality contained on the floppy card. This unique statistical information effects how that player performs in a typical baseball game (i.e., average RBI's, Batting averages, home runs, and so forth). The entire Baseball Game is contained on the companion CD ROM and the players are available to the user only as they have been collected via the floppies. Any player collected on a card may be added to a team file by clicking the "Add to Collection" button. Then via the dual encryption algorithm the player becomes available for play. Reversely a player may be returned to the floppy and encrypted on the CD ROM, thus making them available for trading with a friend. This key collection becomes a user's personalized all star team assembled and played against the computer CD ROM baseball game or with another collector's team via the on-line version of the game. The trading card game is enhanced when a companion CD ROM is used to facilitate the collection, allowing the players to compete. It is not necessary to have the CD ROM to participate in a collection series of baseball cards, they may be viewed and collected as they appear on the floppy.

The interactive game **194** can be played by the user alone or on-line with another computer system shown generally at **200** having a similar CD ROM program, as depicted in FIG. **9**. Shown therein is the computer system **30** of FIG. **1** and another similar computer system shown generally at **30'** operatively connected together via a telephone line **202** and telephone modems **204** and **206** (see U.S. Pat. No 5,273,288 to Teshima et al.). Components of system **30'** corresponding to those of system **30** are assigned the same reference numeral followed by a prime designation.

Preferred collectible items of this invention are baseball cards, as previously described in detail. Of course other items can be adapted and collected, such as "cards" for other sports figures, for actors and actresses, or for historical figures. For example, one embodiment can be an educational game where the users collect and learn about former United States Presidents. Another educational game can comprise collecting former generals and admirals (possibly together with various weapon systems) and the interactive game could reenact famous battles, using the collected soldiers and weapons, either against the computer or on-line (See FIG. **9**) against a fellow collector.

Another collection game **194** involves collecting "monster parts" and assembling them with different attributes or powers. One of the monster parts (or collection items) could be an "essence" card (or "live force vial") which is the part needed to bring the assembled monster to life. The "essence" cards could be the more rare, that is, provided on fewer floppies (**56**, **58**, **60**), and thus more desirable, valuable and

tradeable. With the monster assembled and brought to life and the interactive game actuated, the player's (good) monster can be pitted against the game's (evil) monsters created by an evil doctor. The object of the game would be to access the evil doctor's laboratory and shut off the power. To help the user prevail some of the items collected from floppies can be "weapons," such as monster stitch desolver, iodine revealer for revealing invisible evil monsters, gauze bandages for tieing them up, golden daggers or things to increase the (good) monster's fighting capabilities such as invisibility cartridges or power bolts providing flying powers.

An adult version of the present collection system can be a take-off of the old video arcade peepshows, with the collection items being (pictures of) attractive women. The content played from the floppy on the CD ROM game itself can be audio/visual displays of the collected ladies performing a striptease.

The Multimedia PC Marketing Council, Inc. has developed a second-level multimedia computer specification to encourage the adoption of enhanced multimedia capabilities. This specification is a backwardly compatible superset of the MPC Level 1 Specification, which continues in effect. This Specification defines the minimum system functionality for Level 2 compliance but is not intended as a recommendation for a particular system configuration. The following minimum Multimedia PC Level 2 System requirements are applicable to the present invention:

I. Hardware:

25 MH 486 SX or compatible microprocessor

Four megabytes of RAM (Eight megabytes recommended)

3.5"Floppy drive

Hard drive (160 MB minimum)

Video display resolution of a least 640×480 with 65,536 (64K) colors

Two button mouse

101 key Keyboard (or functional equivalent)

CD-ROM Drive:

- Double-speed with CD-DA outputs (Capable of sustained 300 KB/sec transfer rate)
- No more than 40% of the CPU bandwidth may be consumed when maintaining a sustained transfer rate of 150 KB/sec
- Average seek time of 400 milliseconds or less
- 10,000 hours MTBF
- CD-ROM XA ready (mode 1 capable, mode 2 form 1 capable, mode 2 form 2 capable)
- Multisession capable
- MSCDEX 2.2 driver or equivalent that implements the extended audio APIs
- Subchannel Q support (P, R-W optional)

Audio board:

- 16-bit DAC, Linear PCM sampling; 44.1, 22.05, and 11.025 kHz rate,
- DMA/FIFO buffered transfer capability
- 16-bit ADC, Linear PCM sampling; 44.1, 22.05, and 11.025 kHz rate,
- DMA/FIFO buffered transfer capability; microphone input
- Music synthesizer
- On-board analog audio mixing capabilities
- CD-ROM XA audio capability is recommended
- Support for the ImA adopted ADPCM software algorithm is recommended

Serial port

Parallel port

MIDI I/O port

Joystick port

Headphones or speakers connected to the computer system

II. System Software:

Binary compatibility with Windows 3.0 plus Multimedia Extensions or Windows 3.1.

CD-ROM/Sound Card Audio Cable Standard for MPC Components—Same cable standard as Level 1 (full systems upgrade kits are not required to observe this specification).

To deter copying fin the event the algorithm codes are uncovered) the houses or cases **210** of the floppies **56** are themselves made unique. As shown in FIG. **7**, each has a hologram **212** affixed thereto by adhesive and unique artwork with product names **214** on the front side and a unique serial number **216** printed on the back side as shown in FIG. **8**. Thus, to have the "bootleg" floppy identical to the genuine one, not only would the code keys and programs need to be discovered and copied but also the hologram and serial number. Maximum collectibility, resale value and protection of contents are thereby enhanced.

The floppy disks can be purchased through retail outlets or acquired by trading with fellow collectors. Alternatively, the Collector inserts a plain floppy disk into a kiosk assembly (not shown) which transforms it into a disk usable herein, that is with the program and coded keys. He can choose on the kiosk screen via a main menu the coded keys to be added, or it can be a random selection by the kiosk assembly. Preferably, he would first purchase a blank disk with the hologram and serial number on it. He would enter the serial number into the kiosk, insert the disk which would verify the serial number and write the algorithm, protect the disk and give the corresponding software or the software he chooses.

A "card" or collection item collected can deposit its key **74** into the system file via the encryption/decryption algorithm to await the addition of the companion CD ROM. This takes the chosen key out of circulation and leaves the remaining key (on the floppy disk) for trading purposes. The cards collected individually deposit their key into the system file awaiting the next key in succession to unlock its media. That key can be found on another floppy or the CD ROM. Floppy to floppy they are like a string of puzzle pieces, each floppy containing a portion of the entire puzzle. Keys can be collected in any order. The CD ROM programs can be run disk to disk, floppy to floppy. As the CD media increases its storage capacity, this invention can also be CD to CD, transferring their keys in the same manner as floppies to CD.

The concepts of this invention can be adapted by those skilled in the art for use by equipment other than CD-ROM—floppy disk computers. Examples of other equipment are game cartridges and recently-commercialized memory cards used with HP laptops for example but having no spinning disk to conserve electricity.

U.S. Pat. No. 5,434,464 to Terasima et al shows a video entertainment system with a CD ROM and a pair of separate computer systems. U.S. Pat. No. 5,112,051 to Darling et al. shows an interfacing device for a computer game system. U.S. Pat. No. 4,905,280 to Wiedemer shows a per person use system with video game programming.

From the foregoing detailed description, it will be evident that there are a number of changes, adaptations and modifications of the present invention which come within the province of those skilled in the art. However, it is intended

that all such variations not departing from the spirit of the invention be considered as within the scope thereof as limited solely by the claims appended hereto.

What is claimed is:

1. A computer-based collection system, comprising:

a computer system including a disk drive, a hard drive, a user input and a display system;

a program loaded into said computer system, said program including different locked first and second collection items;

a first disk having a first coded key, said first disk after being inserted into said disk drive being capable of unlocking, using said first coded key, said first collection item into said hard drive; and

a second disk having a second coded key, said second disk after being inserted into said disk drive being capable of unlocking, using said second coded key, said second collection item into said hard drive;

wherein said first coded key after unlocking said first collection piece is rendered inoperative and incapable of unlocking a corresponding first collection item in a similar program.

2. The collection system of claim 1 wherein said program is capable, by actuation of said user input, after said first collection item has been unlocked into said hard drive, of relocking said first collection item relative to said hard drive and rendering said first coded key of said first disk operative and capable of again unlocking said first collection item.

3. The collection system of claim 2 wherein said first disk with said first coded key operative allows said first collection item to be fully displayed on said display system and said first disk with said first coded key inoperative is incapable of fully displaying said first collection item on said display system.

4. The collection system of claim 3 wherein said first collection item when prevented from being fully displayed is displayed with visually dimmer display than when fully displayed.

5. The collection system of claim 4 wherein said computer system, said disk drive and said hard drive define, respectively, a first computer system, a first disk drive and a first hard drive; and further comprising a second computer system including a second disk drive and a second hard drive, a second program similar to said first program and in said second computer system, and wherein said first disk with said first coded key having been rendered operative can be inserted into said second disk drive to unlock using said first coded key a similar first collection item into said second hard drive.

6. The collection system of claim 1 wherein said second coded key after unlocking said second collection piece is rendered inoperative and incapable of unlocking a similar second collection piece in a similar program.

7. The collection system of claim 1 wherein said computer system includes a CD ROM drive operatively connected to said hard drive, and said program is loaded into said computer system by a CD ROM disk into said CD ROM drive.

8. The collection system of claim 7 wherein said first disk includes a first disk program, which includes said first coded key, and which when activated initiates a display, separate from said CD ROM program, on said display system, said display being associated with said first collection item.

9. The collection system of claim 1 wherein said program includes a locked third collection item different from said first and second collection items, and said first disk includes a third coded key adapted to unlock said third collection item into said hard drive.

10. The collection system of claim 1 wherein said first disk includes an identifier on its disk case.

11. The collection system of claim 10 wherein said identifier is a hologram.

12. The collection system of claim 10 wherein said second disk includes a different identifier, different from said identifier of said first disk, on its disk case.

13. The collection system of claim 12 wherein said identifier and said different identifier comprise different serial numbers.

14. The collection system of claim 1 wherein said first and second items form part of a larger set of collection items included in said program.

15. The collection system of claim 1 wherein said user input includes a mouse and a keyboard both operatively connected to said hard drive.

16. The collection system of claim 1 wherein said display system includes a visual display monitor and at least one audio speaker, both operatively connected to said hard drive.

17. The collection system of claim 1 wherein said program includes first and second encryption/decryption algorithms corresponding to said first and second collection items, said first coded key is associated with a complementary first encryption/decryption algorithm complementary to said first encryption/decryption algorithm, and said second coded key is associated with a complementary second encryption/decryption algorithm complementary to said second encryption/decryption algorithm.

18. The collection system of claim 1 wherein said program includes an interactive computer game which a user can play using said user input only after said first and second collection items have been unlocked into said hard drive.

19. The collection system of claim 1 wherein said first disk includes a third coded key, and further comprising a third disk containing said third coded key but not said first and second coded keys.

20. A computer-based collection method, comprising the steps of:

providing a computer system including a disk drive, a hard drive, a CD ROM drive, a user input and a display system;

providing a CD ROM program loaded into said CD ROM drive, the program including different locked first and second collection items;

providing first and second disks having respective first and second coded keys;

inserting the first disk into the disk drive;

instructing the computer system via the user input to unlock the first collection item onto the hard drive using the first coded key of the inserted first disk;

inserting the second disk into the disk drive; and

instructing the computer system via the user input to unlock the second collection item, into the hard drive using the second coded key of the inserted second disk.

21. The collection method of claim 20 wherein the program includes an interactive computer game, and further comprising after said second instructing step and only after a specified series of collection items, including at least the first and second collection items, have been unlocked into the hard drive using respective coded keys contained on disks, playing the computer game using the user input.

22. The collection method of claim 21 wherein said playing includes playing the computer game on-line with another player.

23. The collection method of claim 20 further comprising inserting a third disk having therein a coded key into the disk drive; viewing on the display system a collection item associated with the coded key of the third disk; and without rendering the coded key inoperative, removing the third disk from the disk drive and trading it with another person for at least a fourth disk having a coded key therein.

24. The collection method of claim 20 wherein said first-mentioned instructing step renders the first coded key inoperative and incapable of unlocking a similar first collection piece in a similar program.

25. The collection method of claim 24 further comprising after said firstmentioned instructing step, removing the first disk from the disk drive, and thereafter inserting a disk having an inoperative first coded key into the disk drive, and instructing the computer system to relock the first collection item and render the inoperative first coded key operative and capable of subsequently unlocking the first collection item.

26. The collection method of claim 25 further comprising after said rendering the first coded key operative, inserting the disk into a different computer system and unlocking a corresponding first collection item therein.

27. The collection method of claim 20 further comprising after said second-mentioned instructing step, displaying the first and second collection items on the display system.

28. The collection method of claim 20 wherein said first-mentioned instructing step renders the first coded key inoperative and incapable of unlocking a similar first collection piece in a similar program.

29. The collection method of claim 20 wherein said instructing steps use public key encryption technology.

* * * * *